A Hierarchical Clustering Based Travel Time Estimation Model in a Connected Vehicle Environment

Abdullah Kurkcu
Department of Civil and Urban Engineering & Center for Urban Science + Progress (CUSP), Tandon School of Engineering, New York University (NYU), New York, USA
Email: ak4728@nyu.edu

Kaan Ozbay
C2SMART Center (A Tier 1 USDOT UTC), Department of Civil and Urban Engineering & Center for Urban Science + Progress (CUSP), Tandon School of Engineering, New York University (NYU), New York, USA
Email: kaan.ozbay@nyu.edu

***Abstract*—The Connected Vehicle (CV) technology has the potential to transform driver behavior and will become a promising real-time data source that provides information required to accurately estimate traffic conditions. The information generated by CVs -including speed, position, and acceleration- can be used to analyze, evaluate, and improve the efficiency of the existing transportation infrastructure. In this study, the hierarchical clustering approach based on Wasserstein distances is used to estimate travel times using simulated CV data in an urban setting. The proposed methodology combines segments within a roadway section that have similar speed profiles into clusters and uses these grouped sections to compute the travel time on an individual section. The Basic Safety Messages (BSM) data are simulated from a calibrated traffic model using the Trajectory Conversion Tool (TCA). The generated messages with 5 and 10% market penetration levels are used as input for the clustering based travel time estimation algorithm. The results show that it is possible to accurately estimate travel time using CV data even with lower market penetration levels.**

***Index Terms*—connected vehicles, hierarchical clustering, travel time estimation, performance measures, CV applications**

I. Introduction

Performance measures are defined as indicators that provide system efficiency. In this study, travel time is chosen as the performance measure. For instance, travel time variability is an emerging display of performance progressively used by decision makers in the context of transportation. Furthermore, it is a metric that can be measured effectively using many technologies [1], [2]. Travel time information between specific points is critical information for all transportation agencies and travelers [3]. Accurate estimation of travel times reveals the system performance. Travel time is conventionally detected with fixed infrastructure for ITS operations. With the technological developments, new ways of traffic sensing, computing, and communication methods have become available. Processing a large amount of real-time traffic data has become convenient for traffic operations and traveler information systems. Web-based real-time traffic applications such as Bing Maps, INRIX, Google Maps, and Waze started to release information about the current state of traffic. However, the information provided by CVs is different from web-based real-time traffic applications. The first dissimilarity is that the current real-time traffic web services collect data through a GPS device. Collected data are filtered to remove outliers and are used to provide traffic information such as estimated vehicle speed, and travel time. On the other hand, CVs will have direct access to the present state of the vehicle which provides more accurate information.

The CV platform has the potential to transform the way Americans travel and will supply promising real-time data source that provides information required to estimate traffic conditions on a network. Such information -including speed, position, and acceleration- can be used to understand, evaluate, and improve the effectiveness of the existing transportation infrastructure. The messaging standard was mainly focused on static element messaging through the Dedicated Short Range Communication (DSRC) technology in the United States Department of Transportation (USDOT) Vehicle-Infrastructure Integration (VII) Program [4]. The US messaging protocols involve generation of probe data message (PDM) and basic safety message (BSM). PDM includes periodic snapshots of vehicle position and speed stored in a buffer and transmitted when within transmission range of a roadside equipment (RSE) [5], [6]. BSM consists of vehicle size, position, speed, acceleration, brake system status, etc. at every 0.1 s and either periodically or in the case of special events such as engagement of antilock brakes, starting windshield wipers, changes in temperature, etc. [5]. Most mobility applications in the literature utilize

Manuscript received June 25, 2017; revised December 11, 2017.

doi: 10.18178/jtle.5.2.54-59

BSM protocol to calculate performance measures such as travel time for CV environments. The main purpose of these applications is to utilize frequently collected and transmitted data gathered from connected elements of the system to improve travel efficiency in terms of time and comfort while reducing negative environmental impacts and safety risks. However, the ability to use the CV data relies on the market penetration rate of CVs, the success of the appropriate filtering tools, and the essential traffic conditions. Therefore, it is critical to develop, analyze and implement a comprehensive methodology that considers both the traffic conditions on the accuracy of the estimation of critical transportation measures as well as the effects of market penetration rates of the CV technology.

In this paper, a hierarchical clustering based travel time estimation model using Wasserstein distances is proposed to reduce the error in the assessment of travel time in an urban environment. The proposed algorithm divides selected roadway into cells. Then, the cells with similar speed profiles are clustered to form sections. These sections are then used in the estimation of route travel time along with removing outliers. The rest of this paper is structured as follows. The next section provides a review of the studies utilizing various clustering techniques and their applications in CV environments. The methodology section explains the Wasserstein dissimilarity measure and the proposed clustering approach. Section IV delivers information about the experimental setting, traffic simulation model, and the tools used for this study. Section V provides results by comparing travel time estimation results using the proposed clustering methodology to the probe data vehicle approach. The last section in this study concludes the paper and offers suggestions for future research.

II. Background and Motivation

Clustering is a process of forming virtual groups using nodes that are in close vicinity based on the defined similarity rules. Many different clustering algorithms are used in CV applications in the recent literature. A detailed overview of different clustering algorithms and their uses in the vehicular ad-hoc network (VANET) can be found in [7]. For instance, Maglaras and Katsaros [8] proposed a distributed clustering algorithm to provide the large-scale VANETs to simplify routing operations. Their approach created lesser and more stable clusters on different settings and transmission ranges. Clustering based methodologies have also been used in travel time traffic forecasting [9]-[14] and locating sensors for travel time information [15]-[18]. Bartin, *et al*. [15] showed that vehicle trajectory data could be used to obtain statistically significant travel time estimations at the study location. The discretization of space on the interested routes helped to find the segments where travel time estimation errors were reduced for the given estimation function. These homogeneous segments were found and grouped by using a clustering approach. The placement of point sensors is also studied by Kianfar and Edara [19]. They studied three different clustering techniques including hierarchical, k-means, and Silhouette Measure. Similar to [15], they divided freeway into cells of equal length and clustered cells that have similar speed profiles. The results showed that the hierarchical and k-means clustering algorithms with prior knowledge about the cells produced the best clusters. Assuming each vehicle can become a sensor in a CV environment, placing hypothetical sensors to roadways and finding the optimal estimation method of performance measures become very critical. On board sensors integrated into CVs become the main equipment to sample the traffic condition.

Although the sensor location problem has been investigated by many researchers using clustering techniques, most of them compared average or aggregated measures between nodes to form clusters [15], [16], [19]. However, a hierarchical clustering of histogram data using Wasserstein distances is used in this paper. There are two main clustering techniques that are heavily used in the literature. The first approach considers the prototype of a cluster as an entity which has the same properties of the other elements in the cluster. For instance, if the cluster's barycenter is close to the entity's barycenter, that entity is considered as an element of that cluster. In the second approach, the prototype of a cluster has to explain the variation as well as the characteristics of the other clustered elements [20]. Furthermore, the selection of the distance measure plays a crucial role when performing clustering. Irpino, *et al*. [20] introduced a new metric for the distance measure which extends the Wasserstein metric, and can be easily computed. Histograms are mostly used to represent not only the range of variability but even the inner variability of complex data [21]. The modal data are needed when it is necessary to analyze information about a group of entities [22]. Therefore, comparing two adjacent cells by looking at their speed distributions would support building more homogeneous sections, which in turn leads to better travel time estimations. Most studies in the literature have used a long freeway segment and stable conditions while comparing different travel time estimation methods. However, this approach is not sufficient to capture the inner variation of travel times within the segments, and it may actually overestimate the travel time in some sections and underestimate in others. Furthermore, estimation algorithms are tested only for stable conditions in which most GPS-equipped probe vehicles based estimation methods work fairly well.

III. Research Methodology

The proposed methodology combines segments within a roadway section that have similar speed profiles into clusters. To accomplish this, the section is first discretized into N number of equal length cells as it seen in Fig. 1.

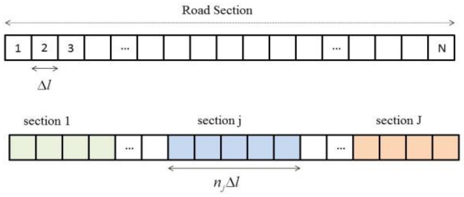

Figure 1. Discretization of the roadway segment using hierarchical clustering method

The selection of Δl depends on the vehicle transmission capacity of connected vehicles. According to Johri, *et al.* [23], given a specific value of the number of lanes (denoted by N) and inter-vehicle gap (denoted by D) on a road segment, the overall transmission capacity is as follows:

$$T = \frac{CU}{(\frac{2RI}{D}+1)N'} \tag{1}$$

where R is the transmission range (default 300 meters), I is the ratio of interference range (default 2 REF), C is the wireless channel capacity (default 27 Mbps which is the maximum rate of the current DSRC), U is the overall capacity utilization ratio (default 0.9), L is the packet length (default 400 bytes). Assuming an inter-vehicle gap of 20 ft., number lanes as 3, and the perfect wireless communication conditions, it was found that 553 ft. is the minimum segment length in which vehicles can transmit messages without exceeding the capacity under dense traffic conditions. Therefore, the cell length is selected to be at least 660 ft. The relationship between the number of cells and the roadway segment length is given by the following:

$$N = L/\Delta l; \quad \Delta l \geq 660\,ft \tag{2}$$

Cells with similar speed patterns can be combined to characterize a segment with homogeneous speed profile to improve the accuracy of critical transportation measure calculations. According to Irpino [22] symbolic data is a multi-valued descriptor with frequency, probability, or weight associated with each of its specific values. Since speed profiles of each cell constructed using the speeds measured at each cell during the data collection period, it is possible to explain them using modality. For example, given a set of units $i \in \Omega$, the modal variable Y is a mapping:

$$Y_i = \{U(i), \pi_i\} \text{ for } i \in \Omega \tag{3}$$

where π_i is an associated nonnegative measure on the domain Y of possible observation values. $U(i) \subseteq Y$ is the support of π_i. In other words, speed histogram data is a proxy for representing the underlying empirical distribution of a continuous variable Y which contains individual speed measurements. Y is divided into a set of consecutive, non-overlapping classes (bins) $U(i)$ associated with π_i weights.

In mathematics, the Wasserstein (or Vaserstein) metric is a distance function used for the evaluation of the convergence of probability distributions on a given metric space X. It is also known as "earth mover's distance" in the literature [24]. Assuming each distribution as a unit amount of "dirt" laid on space X, the Wasserstein (or Kantorovich) distance provides the cost of turning one pile into the other. According to [20], considering a Euclidean norm, the equation used to calculate the Wasserstein distance between distribution functions of Y(i) and Y(j) is:

$$W_2(Y_i, Y_j) = \sqrt{\int_0^1 (F_i^{-1}(t) - F_j^{-1}(t))^2 dt} \tag{4}$$

where F_i and F_j are the cumulative distribution functions and the F_i^{-1} and F_j^{-1} are the quantile functions.

Assuming speed distributions in each cell is represented by histograms, E would be the whole set that contains each cell's speed distribution data described by interval variables. The prototype of the cluster C_j $(j = 1, ..., J)$ is represented by a vector $G = (G_1, ..., G_J)$ where G_j is a histogram. J is the number of clusters, and it varies from 2 to N. In other words, each cluster and cluster prototypes are also represented as histogram data. The clustering algorithm tries to find a partition $P = (C_1, ..., C_J)$ of E in J clusters. The partitioning criterion that is locally minimized for the clustering algorithm is defined as:

$$\gamma(P,G) = \sum_{j=1}^{J} \sum_{i \in C_j} W_2^2(Y_i, G_j) \tag{5}$$

where $W_2^2(Y_i, g_k)$ is a Wasserstein distance measure between objects and $Y_i \in C_j$ the class prototype G_j of C_j.

The clustering algorithm groups similar cells together. The number of groups can be varied from 2 to N. After clustering; only successive cells are grouped together to form detection sections. Such sections are then used in travel time calculation. This step makes sure that cells that are only physically next to each other are clustered. To investigate the accuracy of the proposed approach, two criteria namely, mean absolute percentage error (MAPE) and root mean square error (RMSE) are calculated.

$$RMSE = \left[\frac{1}{n}\sum_{t=1}^{n}\left(GTTT_t - ETT_t\right)^2\right]^{1/2} \tag{6}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|GTTT_t - ETT_t|}{GTTT} \times 100\% \tag{7}$$

where GTTT and ETT are ground truth travel time and the estimated travel time respectively, and n is the number of 5-minute periods.

IV. Simulation Model

To validate the hierarchical clustering algorithm, the microscopic traffic simulation software PARAMICS is used to model urban traffic. The downtown Brooklyn area of New York is selected for the case study. The final traffic simulation model consists of 36 intersections, 22 traffic signals, 19 traffic zones, and 16.35 miles of roadway. The actual properties of roadway links such as the signal timing, length, lane width, number of lanes, and speed limit are also encoded in the network file. The traffic simulation model is calibrated for the AM peak period (7-10AM) using the turning movement counts collected at each intersection and one way travel time from Tillary Street to Grand Army Plaza (Southbound). Fig. 2 below shows the network location and the generated traffic simulation model in PARAMICS. In addition, a traffic incident which occurred at 8:15 AM is simulated for 30 minutes to reflect the non-recurrent traffic conditions. The simulated incident is a broken down vehicle and it blocks the right most lane. Trajectory data are collected every 0.1 seconds in the simulation for an hour between 8-9 AM. The 7632 ft. section between Tillary Street and Grand Army Plaza is selected for further investigation.

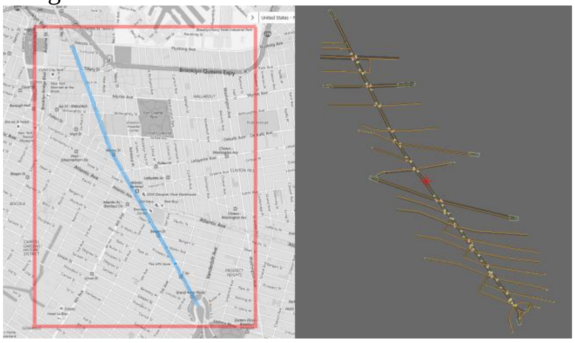

Figure 2. Study location

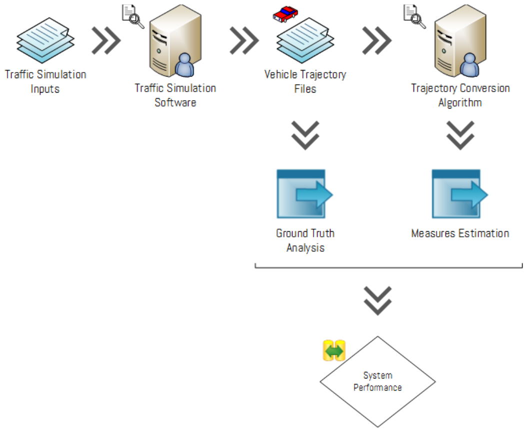
Traffic Simulation Inputs
Traffic Simulation Software
Vehicle Trajectory Files
Trajectory Conversion Algorithm
Ground Truth Analysis
Measures Estimation
System Performance

Figure 3. TCA tool - off-line mode of operation [25]

Once the simulation network is calibrated, CV messages can be generated using software called the Trajectory Conversion Algorithm (TCA) built by NYU UrbanMITS Transportation group and Noblis [25]. This software is intended to test diverse scenarios for generating, transferring, and storing CV information. The developed open-source the TCA tool is one of the vital features of the BSM Data Emulator, and it can work in an on-line or off-line mode. The on-line mode generates CV messages while the traffic simulation is running. On the other hand, the offline mode which will use simulated vehicle trajectories and convert them to CV messages. Fig. 3 graphically illustrates the approach for trajectory conversion to CV messages. Using the off-line mode of the TCA tool, trajectory data collected after running the simulation are converted into BSM CV messages with 5% and 10% market penetration levels to check the accuracy of travel time estimation accordingly. Traffic simulation model calibration remains a difficult and time-consuming task. Lack of data related to the CV applications and the parameters that must be calibrated for each model making this task particularly complex. Therefore, the latency and drop rate of messages are assumed to be "0" for simplicity.

V. Experimental Results

This section presents results comparing the proposed approach with the travel time estimation method using probe vehicle BSM messages. Given the stochastic nature of microscopic simulations, several repetitions of the same scenario with different random seeds should be undertaken. Multiple runs are always desirable with calibrated and validated stochastic microscopic simulation models. Therefore, CV messages are generated using trajectory data from 5 different runs with different seeds. It has been detected that the average error rate in estimating travel times does not change more than 5% after the 5th run. Fig. 4 shows the cluster results after analyzing the first 5 minutes of speed data received by vehicles.

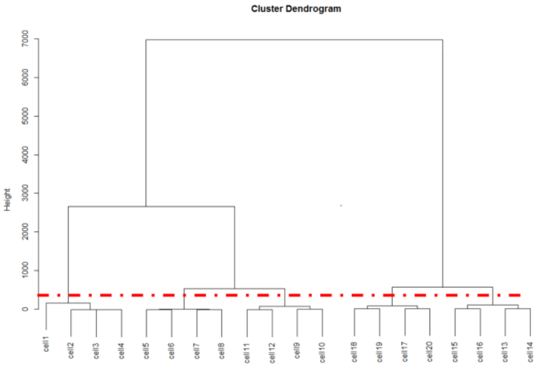

Figure 4. Clustering Results for the first 5-minute Period

Fig. 5 shows the location of the clusters for the first 5-minute period at 10% market penetration as an example. The number of clusters changes in the each 5-min period and it is decided by the height calculated by the dissimilarity measure. If the joining height of two clusters is twice more than the existing merged height, the tree is cut by that static value which defines the number of clusters. The red line in Fig. 4 shows the tree cut. For example, 5 clusters are formed for the first 5-minute period at 10% market penetration level.

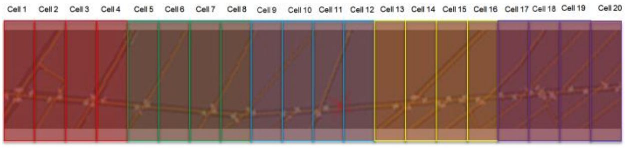

Figure 5. Cluster Locations

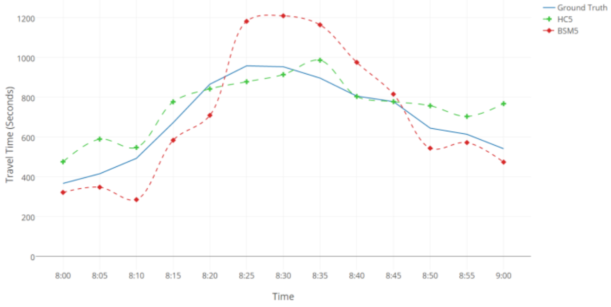

Figure 6. Travel time comparisons at 5% market penetration level

Fig. 6 below illustrates the estimation results for 5% market penetration level. All the results using BSM that are reported in this section are the average metrics of the five simulation runs conducted. The green line is the estimated travel time with the hierarchical clustering model, the red line is the estimated travel time with GPS probe data sampled every 0.1 seconds, and the blue one is the ground truth travel time data. It has been detected that when the incident took place, the accuracy of the estimated travel time with GPS probe data reduces dramatically. After the incident, the accuracy of the estimated travel time improves.

Similar behavior has also been observed for the 10% market penetration level as it can be seen from Fig. 7. Although the GPS probe data poorly estimated the travel time during the incident, the overall accuracy of the estimation is improved with additional probe vehicles. The hierarchical clustering method again performed better than the probe data, and its accuracy is also improved by 17% with the higher market penetration level.

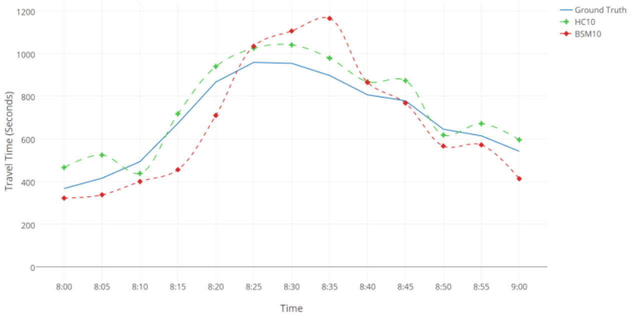

Figure 7. Travel time comparisons at 10% market penetration level

The estimation results for each 5 minute period at both market penetration levels are summarized in Table I. As it can be seen from the table, the clustering method outperformed the probe vehicle approach at both market penetration levels. Although the RMSE for the clustering method for 10% market penetration level is 133.5 which higher than the RMSE values for 5% penetration level, it can better capture the overall distribution of the travel time before, during and after the incident.

TABLE I. Comparison of the Results of the Hierarchical Clustering with Probe Vehicle Approach

Approach	RMSE	MAPE
BSM Probes (5%)	133.2	20.7%
Hierarchical Clustering (5%)	85.07	17.4%
BSM Probes (10%)	108.1	17.4%
Hierarchical Clustering (10%)	133.5	12.6%

VI. Conclusion and Discussion

In this study, the hierarchical clustering method based on Wasserstein distances is used to estimate travel time on the given route in an urban setting. Simulated data from a calibrated traffic model are used to generate BSM messages. These messages generated at 5% and 10% market penetration levels are used as an input for the travel time estimation algorithm. The results showed that it is possible to accurately estimate travel time using CV data even with lower market penetration levels.

The proposed methodology outperformed the traditional GPS-equipped probe vehicle-based travel time estimation methodology. At 5% market penetration level, the MAPE for the clustering and probe vehicle-based travel time estimation methods are 17.4% and 20.7% respectively. When the market penetration level is increased to 10%, the MAPE for the clustering and probe vehicle-based travel time estimation method become 12.6% and 17.4% respectively. The simulated incident is a vehicle breakdown blocking one lane. Thus, all lanes may not have similar levels of queuing and speed reduction. If the sampled vehicle is stuck behind the incident, it may lead to poor travel time estimations. Therefore travel time estimation could be higher than ground truth as the “average” sampled vehicle data does not necessarily reflect conditions for each vehicle. The essential value of the approach is to create homogeneous sections having the statistically significant amount of vehicles to sample data for more reliable estimations. Furthermore, the approach could be applied in real-time since the data processing, and clustering takes less than one minute to execute. However, appropriate and meaningful criteria derived from the historical data have to be carefully chosen before the algorithm is applied to a new section. The scenarios with more market penetration levels, different roadway types, and messaging protocols will be tested in a future study.

References

[1] E. Morgul, H. Yang, A. Kurkcu, K. Ozbay, B. Bartin, C. Kamga, *et al.*, “Virtual sensors: Web-Based real-time data collection methodology for transportation operation performance analysis,” *Transportation Research Record: Journal of the Transportation Research Board*, pp. 106-116, 2014.

[2] D. D. Puckett and M. J. Vickich, “Bluetooth®-based travel time/speed measuring systems development,” *High Occupancy Vehicle Lanes*, 2010.

[3] B. Bartin and K. Ozbay, “Determining the optimal configuration of highway routes for real-time traffic information: A case study,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, pp. 225-231, 2010.

[4] USDOT. (2015, May 06). DSRC: The Future of Safer Driving. [Online]. Available: http://www.its.dot.gov/factsheets/dsrc_factsheet.htm

[5] H. Park, A. Miloslavov, J. Lee, M. Veeraraghavan, B. Park, and B. Smith, “Integrated traffic-communication simulation evaluation environment for IntelliDrive applications using SAE J2735 message sets,” *Transportation Research Record: Journal of the Transportation Research Board*, pp. 117-126, 2011.

[6] S. O. A. E. S. s. J2735, *Dedicated Short Range Communications (DSRC) Message Set Dictionary*, ed, November 2009.

[7] S. Vodopivec, J. Bešter, and A. Kos, “A survey on clustering algorithms for vehicular ad-hoc networks,” in *Proc. 35th International Conference on Telecommunications and Signal Processing (TSP)*, 2012, pp. 52-56.

[8] L. A. Maglaras and D. Katsaros, “Distributed clustering in vehicular networks,” in *Proc. IEEE 8th International Conference on Wireless and Mobile Computing,* Networking *and Communications (WiMob)*, 2012, pp. 593-599.

[9] X. Zhang and J. A. Rice, “Short-term travel time prediction,” *Transportation Research Part C: Emerging Technologies*, vol. 11, pp. 187-210, 2003.

[10] M. Yildirimoglu and N. Geroliminis, “Experienced travel time prediction for congested freeways,” *Transportation Research Part B: Methodological*, vol. 53, pp. 45-63, 2013.

[11] M. Ramezani and N. Geroliminis, “On the estimation of arterial route travel time distribution with Markov chains,” *Transportation Research Part B: Methodological*, vol. 46, pp. 1576-1590, 2012.

[12] Z. Ma, H. N. Koutsopoulos, L. Ferreira, and M. Mesbah, "Estimation of trip travel time distribution using a generalized Markov chain approach," *Transportation Research Part C: Emerging Technologies*, vol. 74, pp. 1-21, 2017.

[13] L. Kisgyörgy and L. R. Rilett, "Travel time prediction by advanced neural network," *Periodica Polytechnica. Civil Engineering*, vol. 46, p. 15, 2002.

[14] J. Yu, G. L. Chang, H. Ho, and Y. Liu, "Variation based online travel time prediction using clustered neural networks," in *Proc. 11th International IEEE Conference on Intelligent Transportation Systems*, 2008, pp. 85-90.

[15] B. Bartin, K. Ozbay, and C. Iyigun, "Clustering-based methodology for determining optimal roadway configuration of detectors for travel time estimation," *Transportation Research Record: Journal of the Transportation Research Board*, pp. 98-105, 2007.

[16] M. Gentili and P. B. Mirchandani, "Locating vehicle identification sensors for travel time information," in *Applications of Location Analysis*, ed: Springer, 2015, pp. 307-327.

[17] H. Yang, K. Ozbay, and K. Xie, "Improved travel time estimation for reliable performance measure development for closed highways," *Transportation Research Record: Journal of the Transportation Research Board*, pp. 29-38, 2015.

[18] M. Asudegi and A. Haghani, "Optimal number and location of node-based sensors for collection of travel time data in networks," *Transportation Research Record: Journal of the Transportation Research Board*, pp. 35-43, 2013.

[19] J. Kianfar and P. Edara, "Optimizing freeway traffic sensor locations by clustering global-positioning-system-derived speed patterns," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, pp. 738-747, 2010.

[20] A. Irpino, R. Verde, and Y. Lechevallier, "Dynamic clustering of histograms using wasserstein metric," in *COMPSTAT*, 2006, pp. 869-876.

[21] L. Billard and E. Diday, "From the statistics of data to the statistics of knowledge: Symbolic data analysis," *Journal of the American Statistical Association*, vol. 98, pp. 470-487, 2003.

[22] A. Irpino and R. Verde, "A new wasserstein based distance for the hierarchical clustering of histogram symbolic data," in *Data Science and Classification*, ed: Springer Berlin Heidelberg, 2006, pp. 185-192.

[23] R. Johri, J. Rao, H. Yu, and H. Zhang, "A multi-scale spatiotemporal perspective of connected and automated vehicles: applications and wireless networking," *IEEE Intelligent Transportation Systems Magazine*, vol. 8, pp. 65-73, 2016.

[24] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, pp. 99-121, 2000.

[25] USDOT. (2015). *TCA 2.3.3*. [Online]. Available: https://www.itsforge.net/index.php/community/explore-applications#/38/67